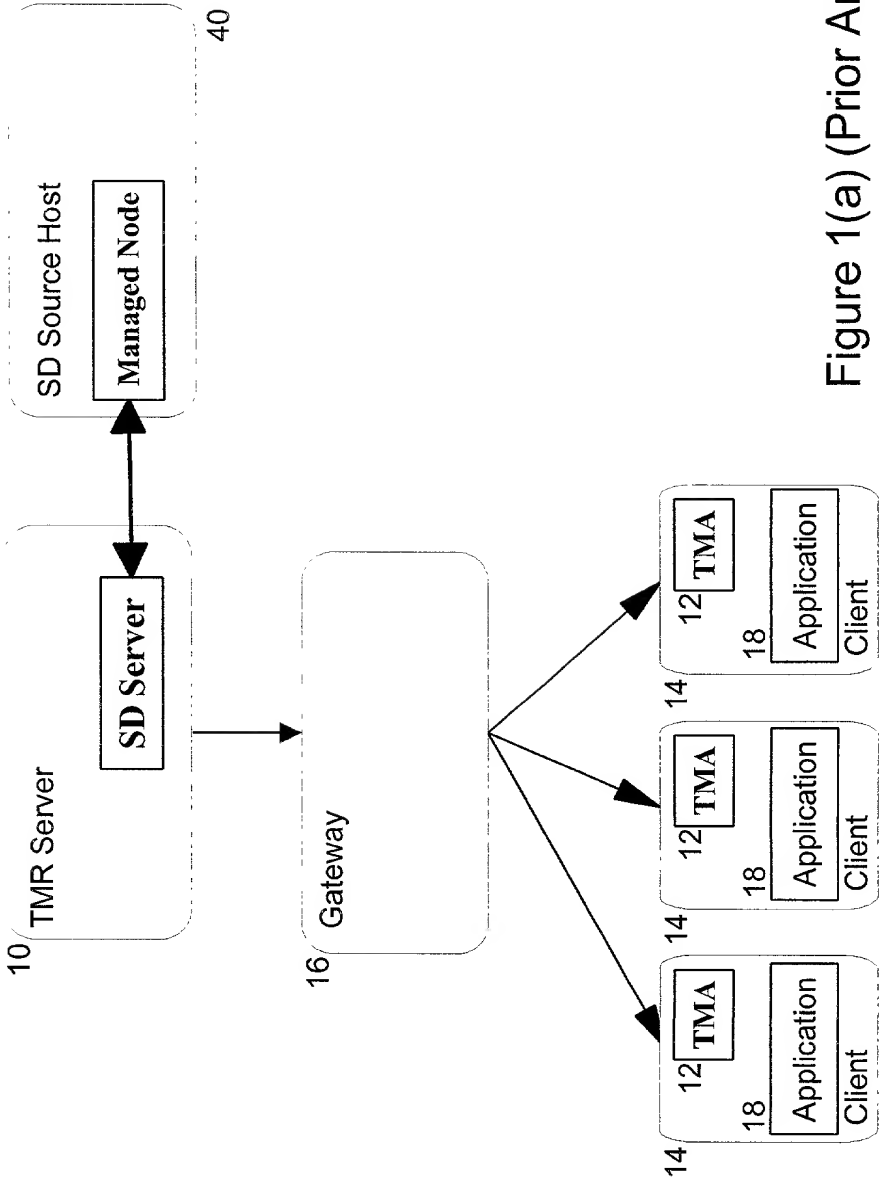Figure 1(a) (Prior Art)
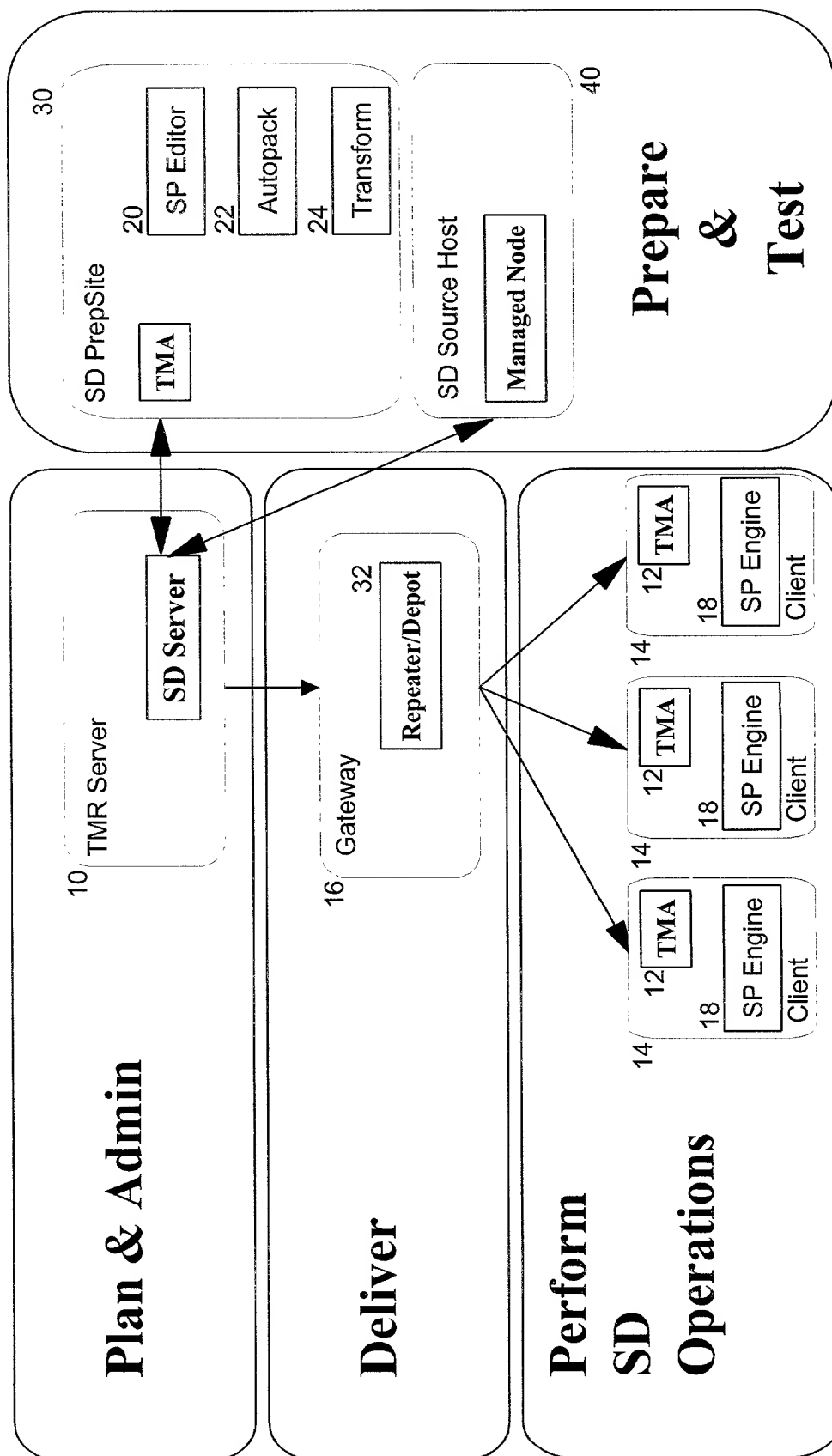
Figure 1(b)
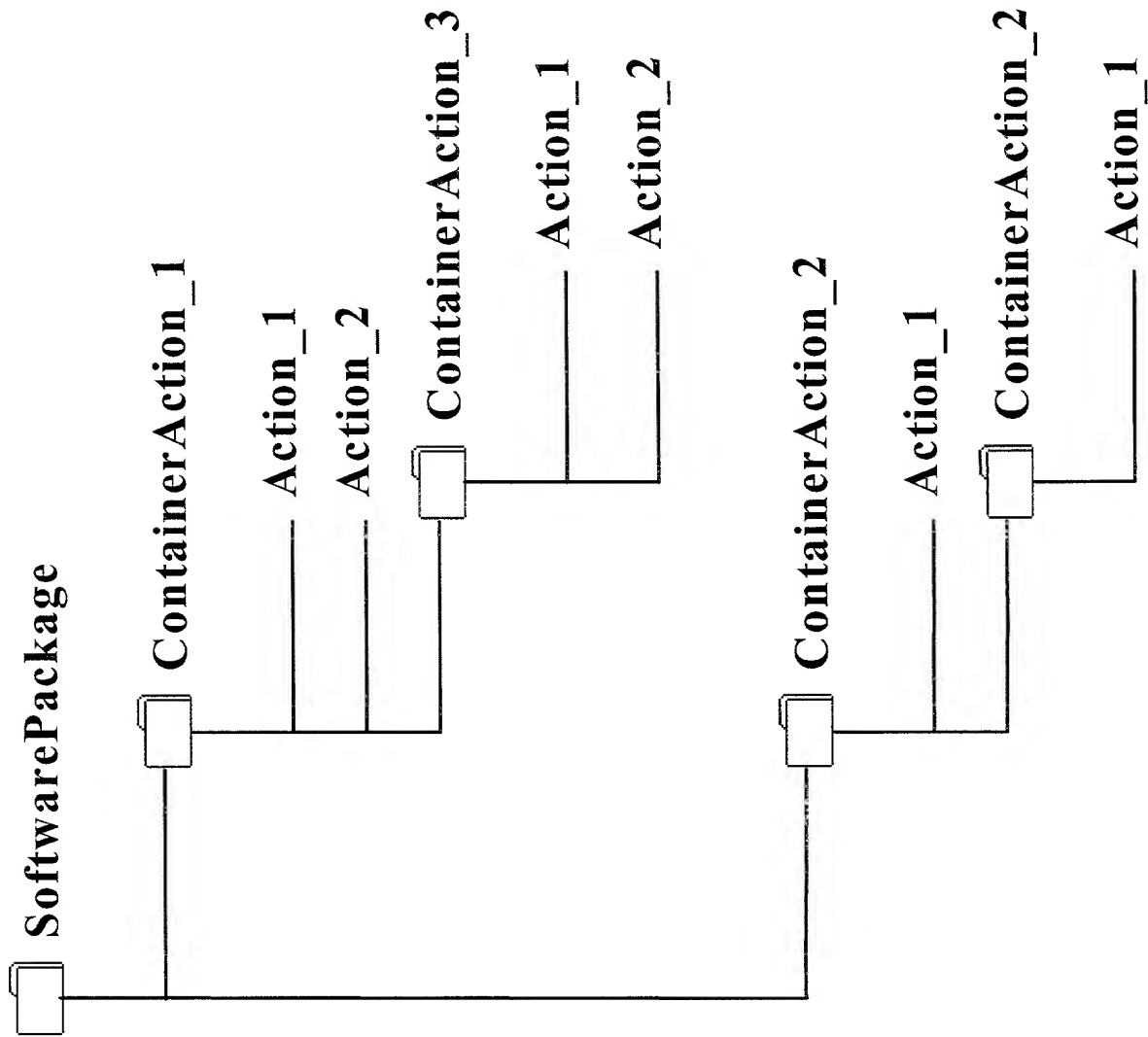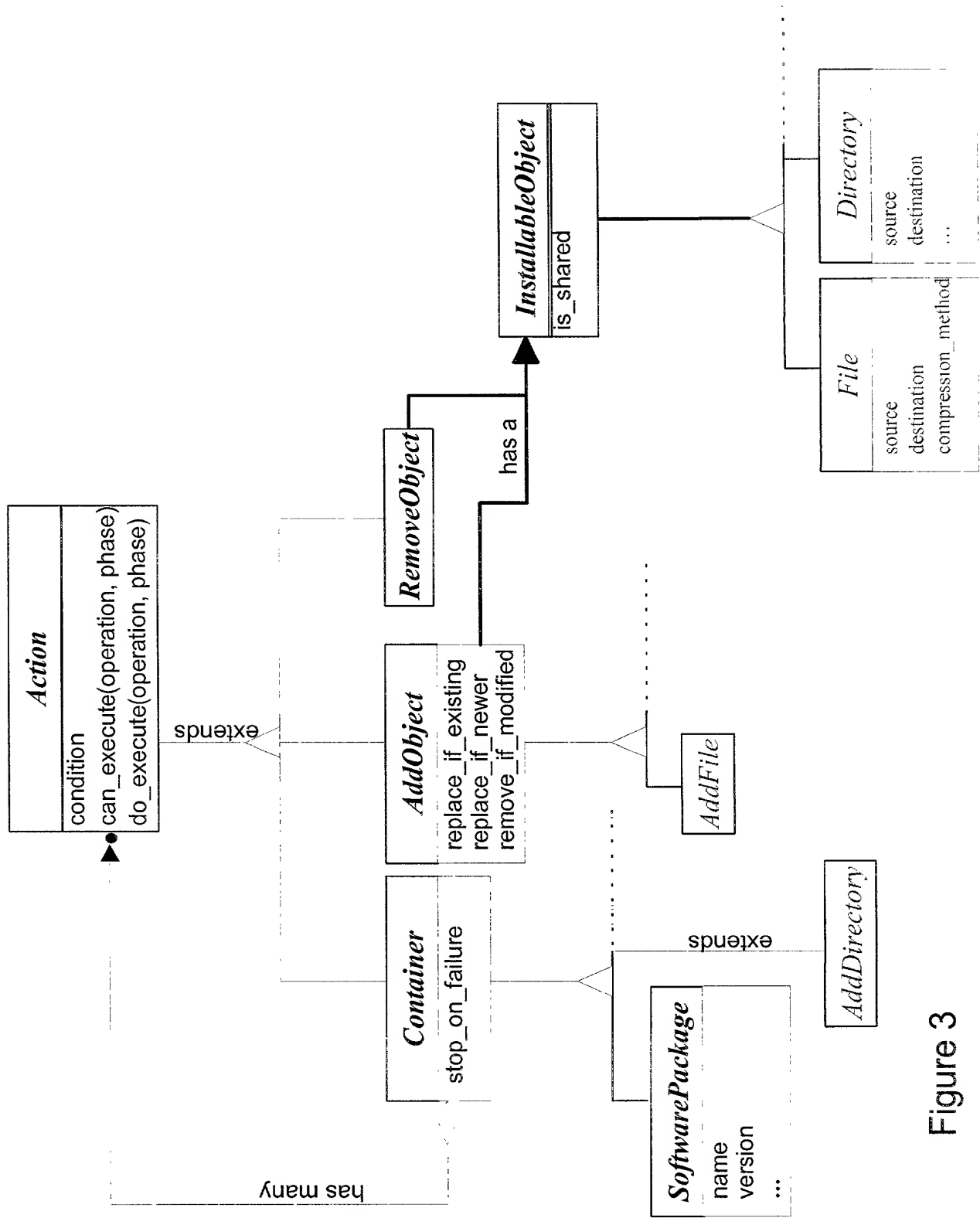
Figure 2

Figure 3
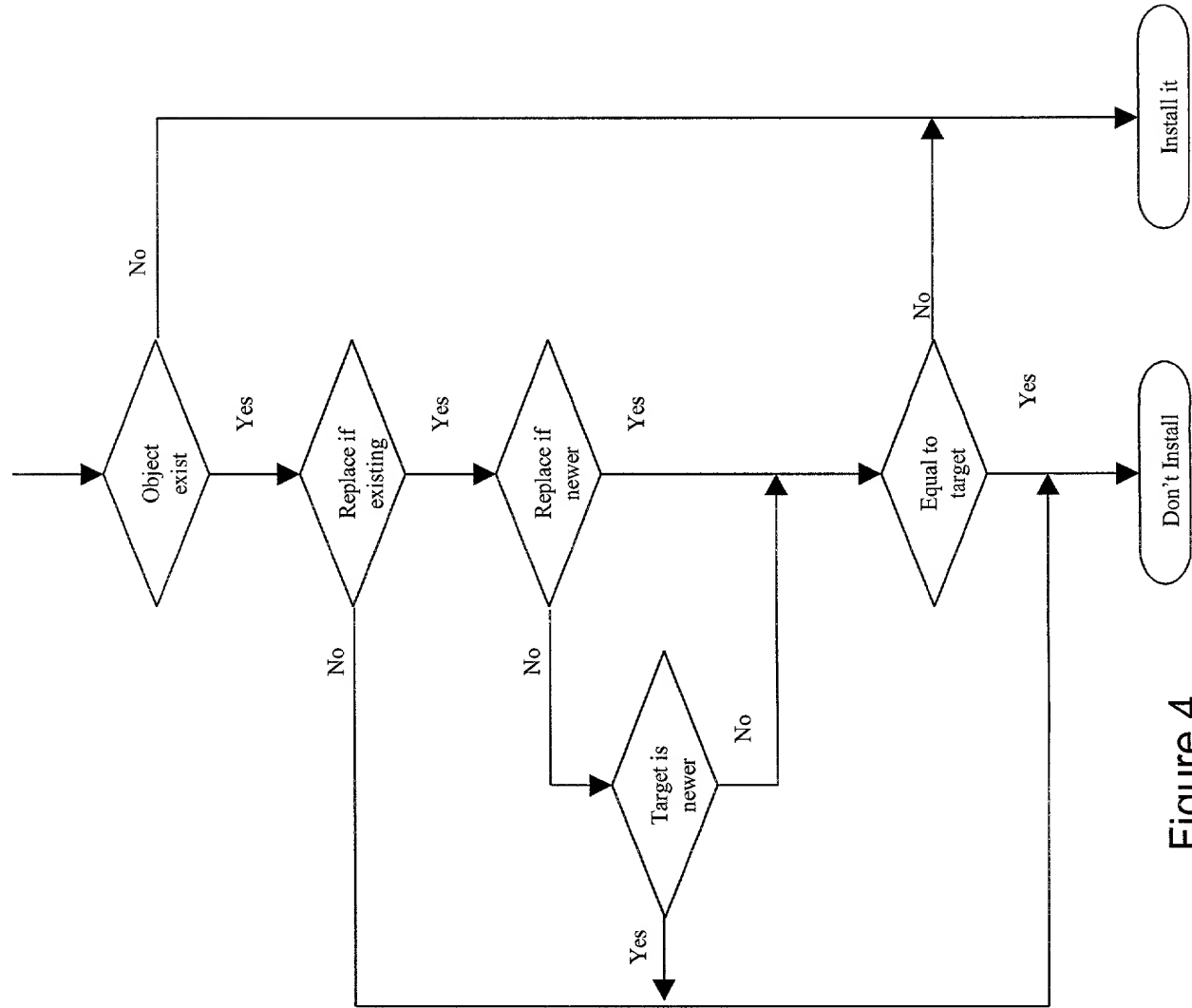
Figure 4
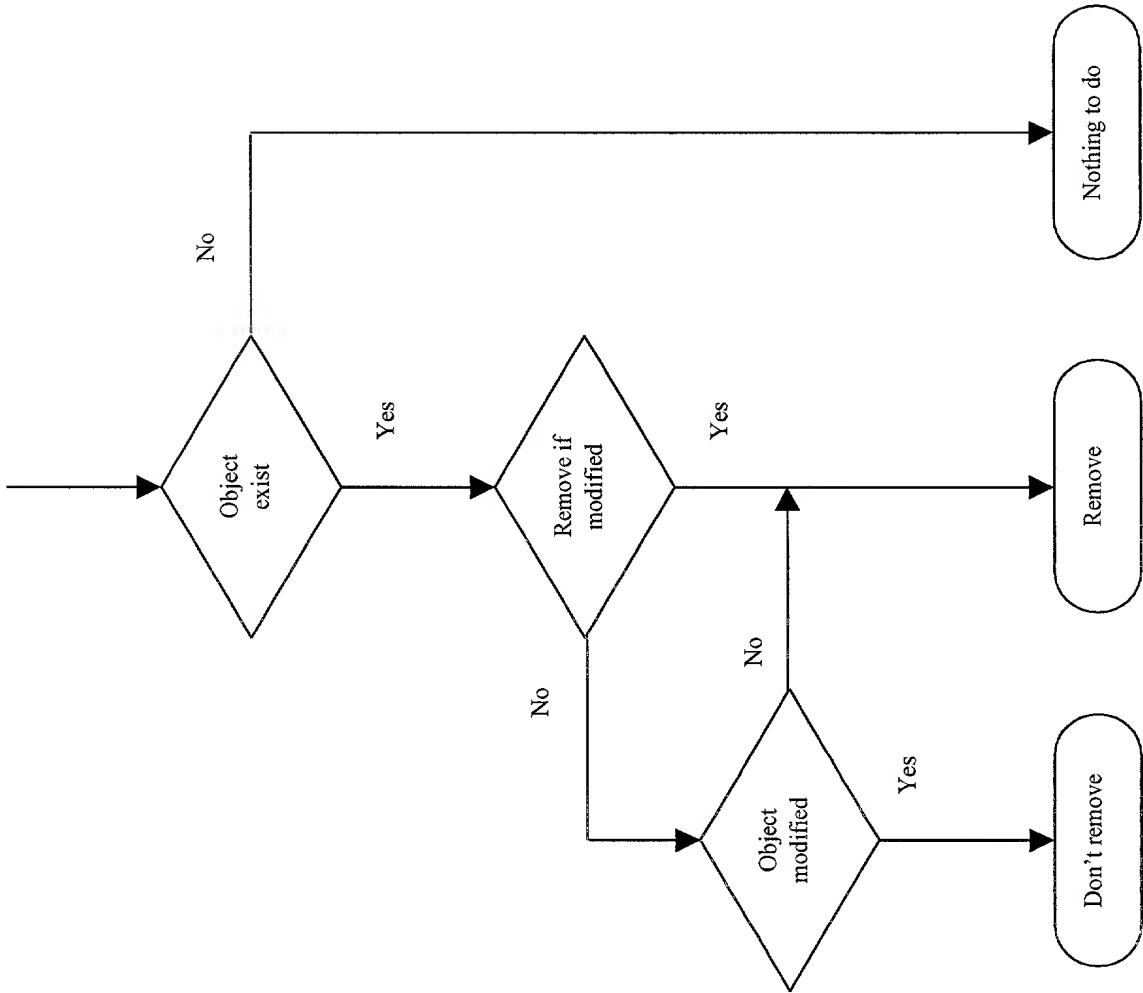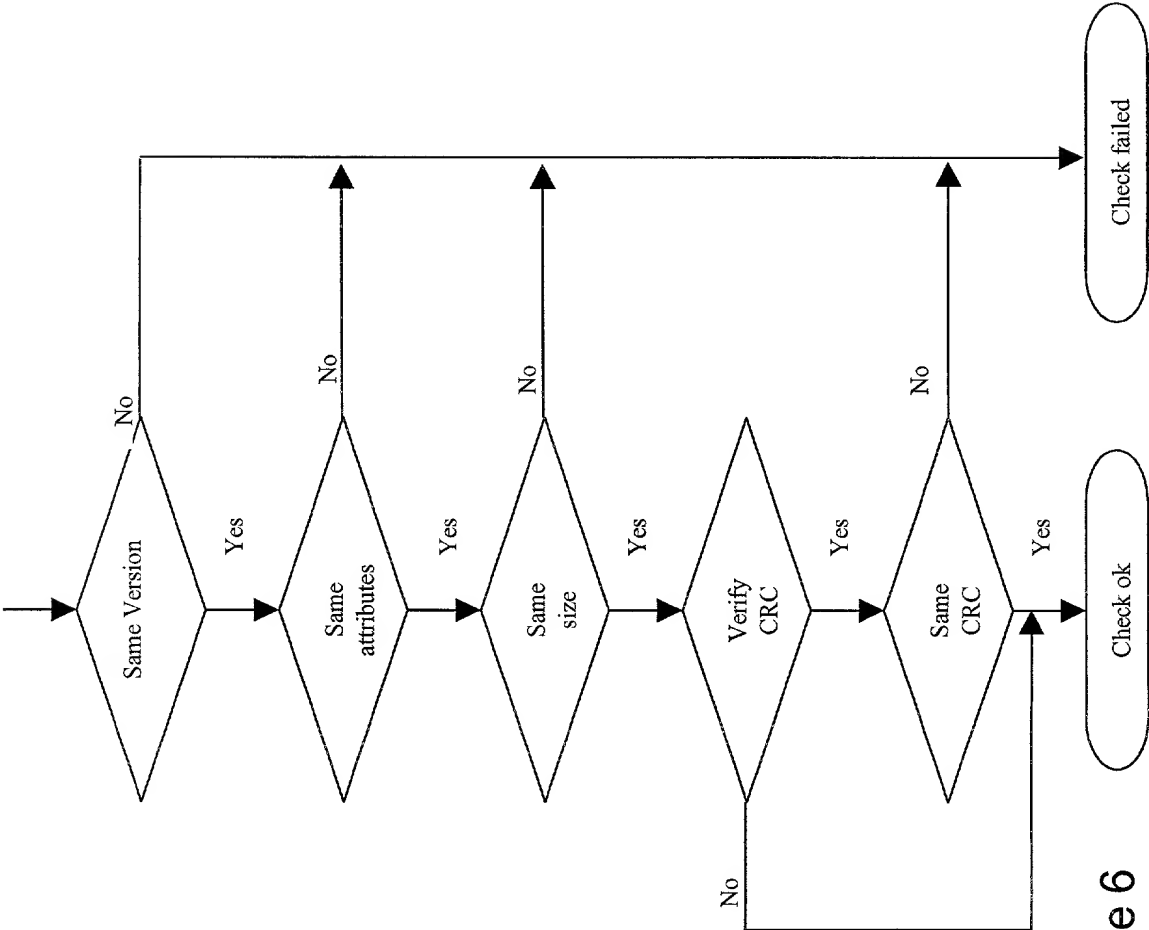
Figure 5

Figure 6

Object related actions	Registry Key and Value
	Shell Folder and Shortcut
	INI file Section and Item
	NT Service
	Profile Item
	Desktop Object, Folder, Program, Shadow
	Directory and File
	Text file line, command, token
	Symbolic link

System actions and Checks	Restart
	Check disk space

Program actions	Execute user program
	Execute MS Setup program
	Execute InstallShield program
	Execute CID Program

Figure 7

add directory c:/Target

add file Readme.txt

add file Config.txt

add directory Bin

add file Start.exe

add file Stop.exe

add win_registry_key HKEY_LOCAL_MACHINE\SOFTWARE\Tivoli

add win_registry_value InstallDir

add win_registry_key SoftwareDistribution

add win_registry_value Version

package

Figure 8

Action

Container — AddObject — AddObject/RemoveObject — 0...N — Container — #children — Action

name	stop_on_failure	replace_if_existing replace_if_newer remove_if_modified	is_shared + InstallableObject attributes	condition	attribute	value	child list ID	#children	Action

Figure 9

Signature of SPD
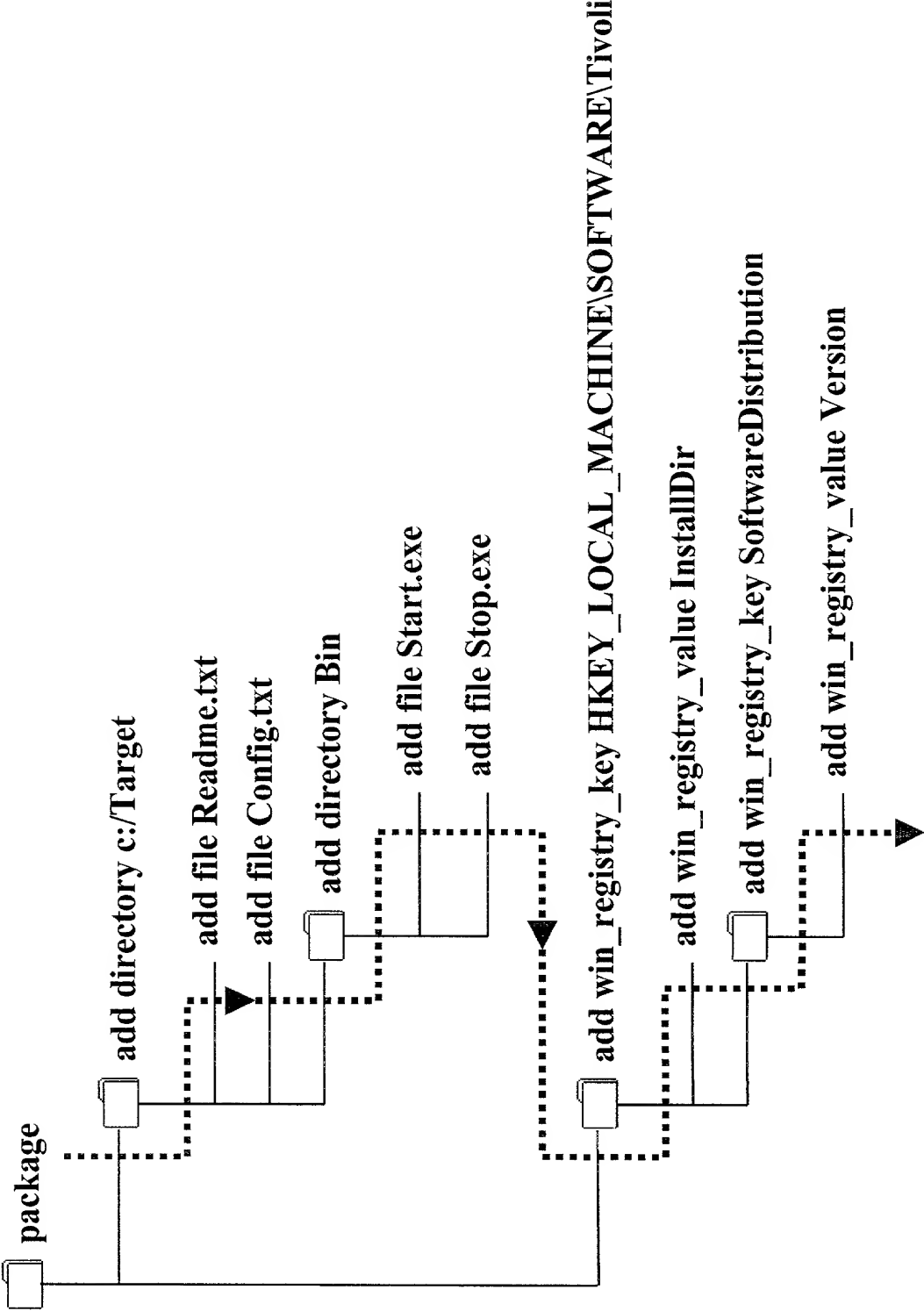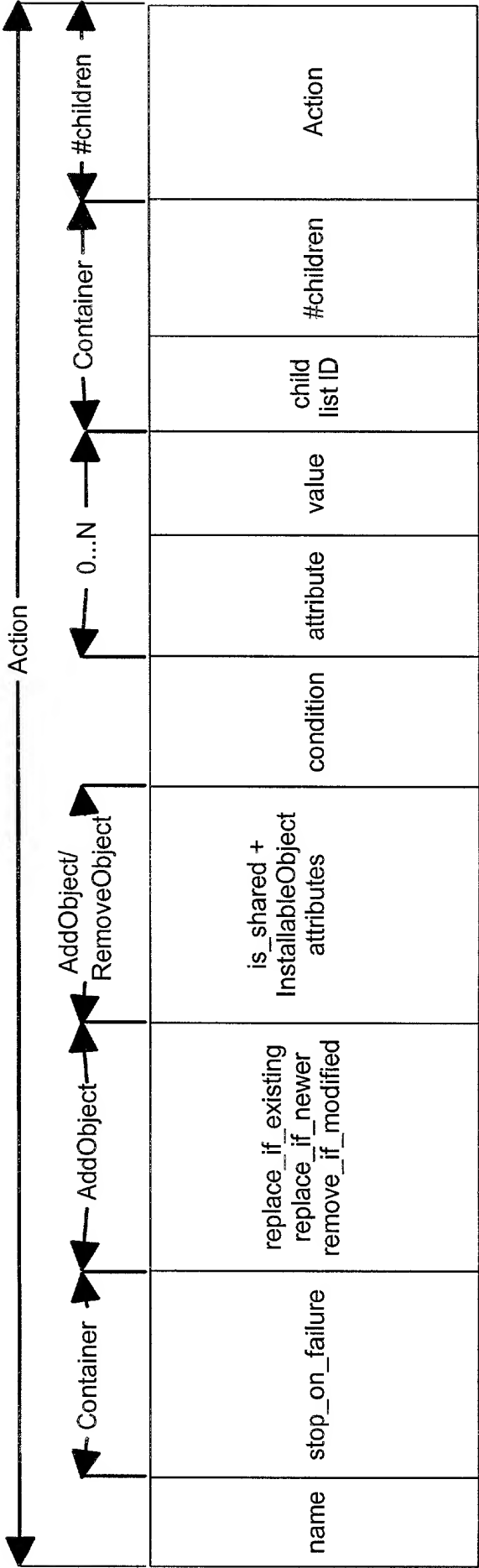
Container stanza

Attribute value pair

```
"TME 10 Software Package v4.0 - SPDF"
package
    name    = "Software Package"
    version = "1.0"

check_disk_space
    volume = C:,25M
    volume = D:,35M
end

generic_container
    condition = "$(os_name) == Windows_NT"

    add win_registry_key

        parent_key = HKEY_LOCAL_MACHINE\SOFTWARE
        key = "Registry Key"

        win_registry_key
            key = "Nested Registry Key"
        end
    end
end
end
```

Figure 10

Figure 11

SPB

SPD

SP

Build SPB from SPD

Export SPB to SPD

Export SPB to SP

Build SPB from SP

Create SP from SPD

Export SP to SPD

```
...
add directory
    location     = c:/Winnt/System32/
    destination  = c:/Winnt/System32/

    file
        name = MyApp.DLL
    end
end

add directory
    location     = c:/MyApp
    destination  = c:/MyApp
    descend_dirs = y
end
...
```
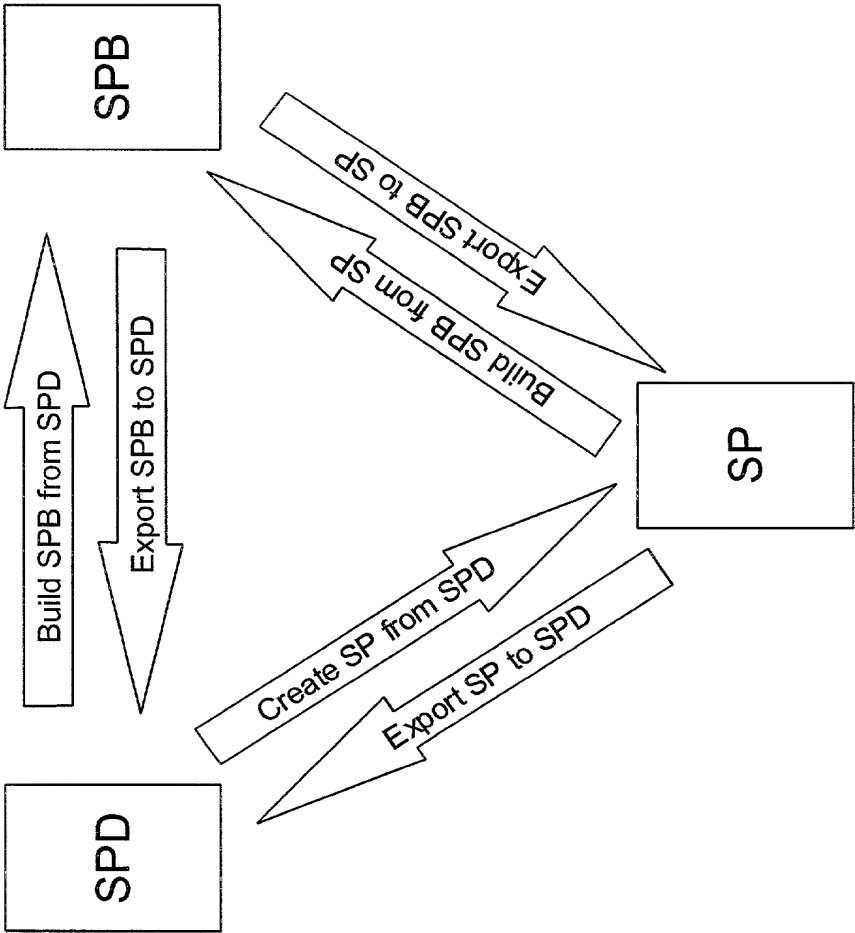
Figure 12(a)

```
...
add directory
    location     = c:/Winnt/System32/
    destination  = $(system_dir)

    file
        name = MyApp.DLL
    end
end

add directory
    location     = c:/MyApp
    destination  = $(target_dir)
    descend_dirs = y
end
...
```

Figure 12(b)

```
...
add directory
    condition = "$(os_name) LIKE 'Win*'"
    location    = c:/winnt/system32
    destination = $(system_dir)
    file
        name = MyApp.DLL
    end
end
add directory
    condition = "$(os_name) == SunOS OR $(os_name) == AIX"
    location    = c:/unix/
    destination = $(home_path)
    file
        name    = MyApp.a
    end
end
...
```

Figure 12(c)

```
...
add directory

  location     = c:/MyApp

  destination  = $(target_dir)

  file

    name = MyApp.CFG

    substitute_variables = y

  end

end

...
```
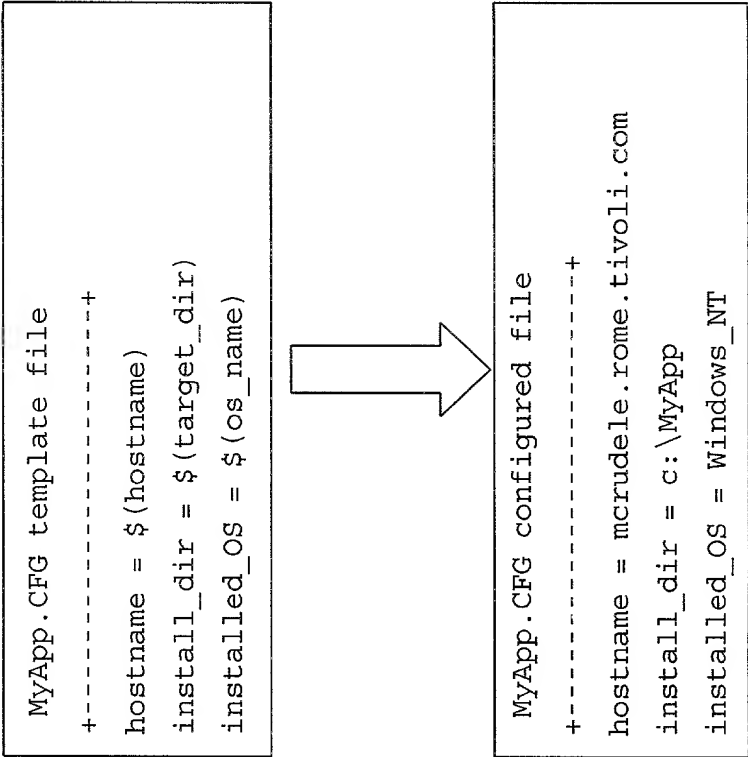
Figure 13(a)

```
+------------------------------------+
| MyApp.CFG template file            |
| +--------------------------------+ |
|                                    |
| hostname = $(hostname)             |
|                                    |
| install_dir = $(target_dir)        |
|                                    |
| installed_OS = $(os_name)          |
+------------------------------------+
```

```
+----------------------------------------+
| MyApp.CFG configured file              |
| +------------------------------------+ |
|                                        |
| hostname = mcrudele.rome.tivoli.com    |
|                                        |
| install_dir = c:\MyApp                 |
|                                        |
| installed_OS = Windows_NT              |
+----------------------------------------+
```

Figure 13(b)

Source Host

TMR Server
running Spider Web Server

Retrieve SPB

Send SPB

Request SP

Web browser

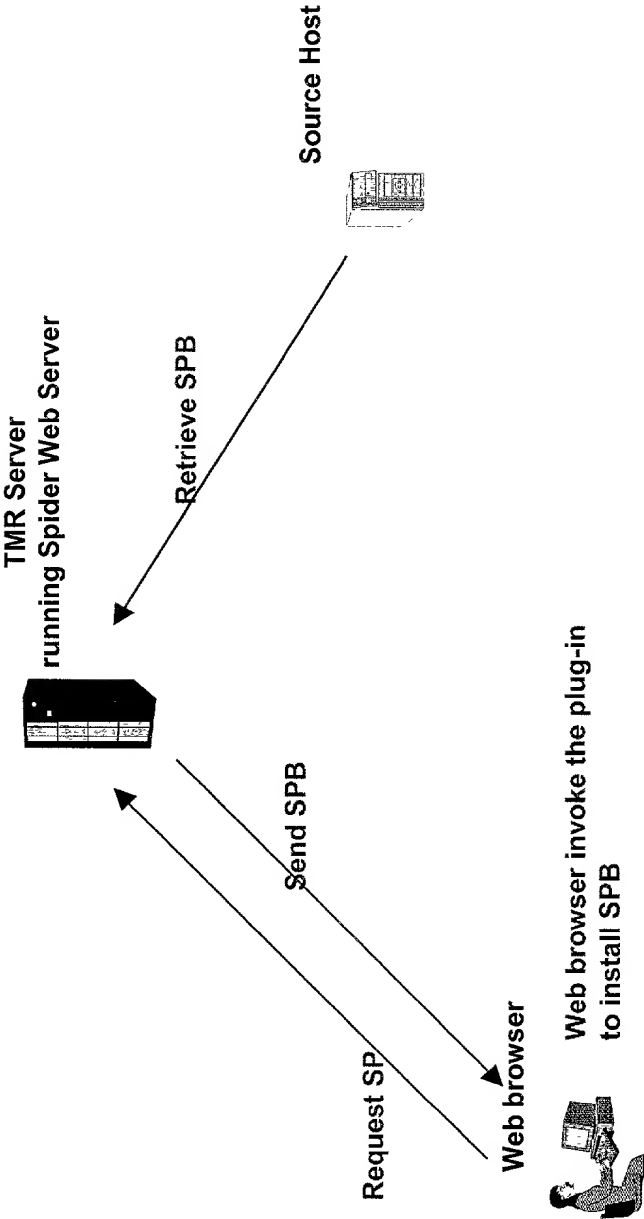Web browser invoke the plug-in
to install SPB

Figure 14

wdcrtsp	Create a SPB file from a SPD file
wdexptsp	Export a SPB file into SPD file
wdbldspb	Build a SPB file from a SP
autopack	Run AutoPack snapshots/differences
wdinstsp	Install a SPB
wdrmvsp	Remove a SP
wdcmmtsp	Commit a SP
wdundosp	Undo a SP Install/Remove
wdacptsp	Accept an undoable Install/Remove
wdversp	Verify the state of a SP

Figure 15

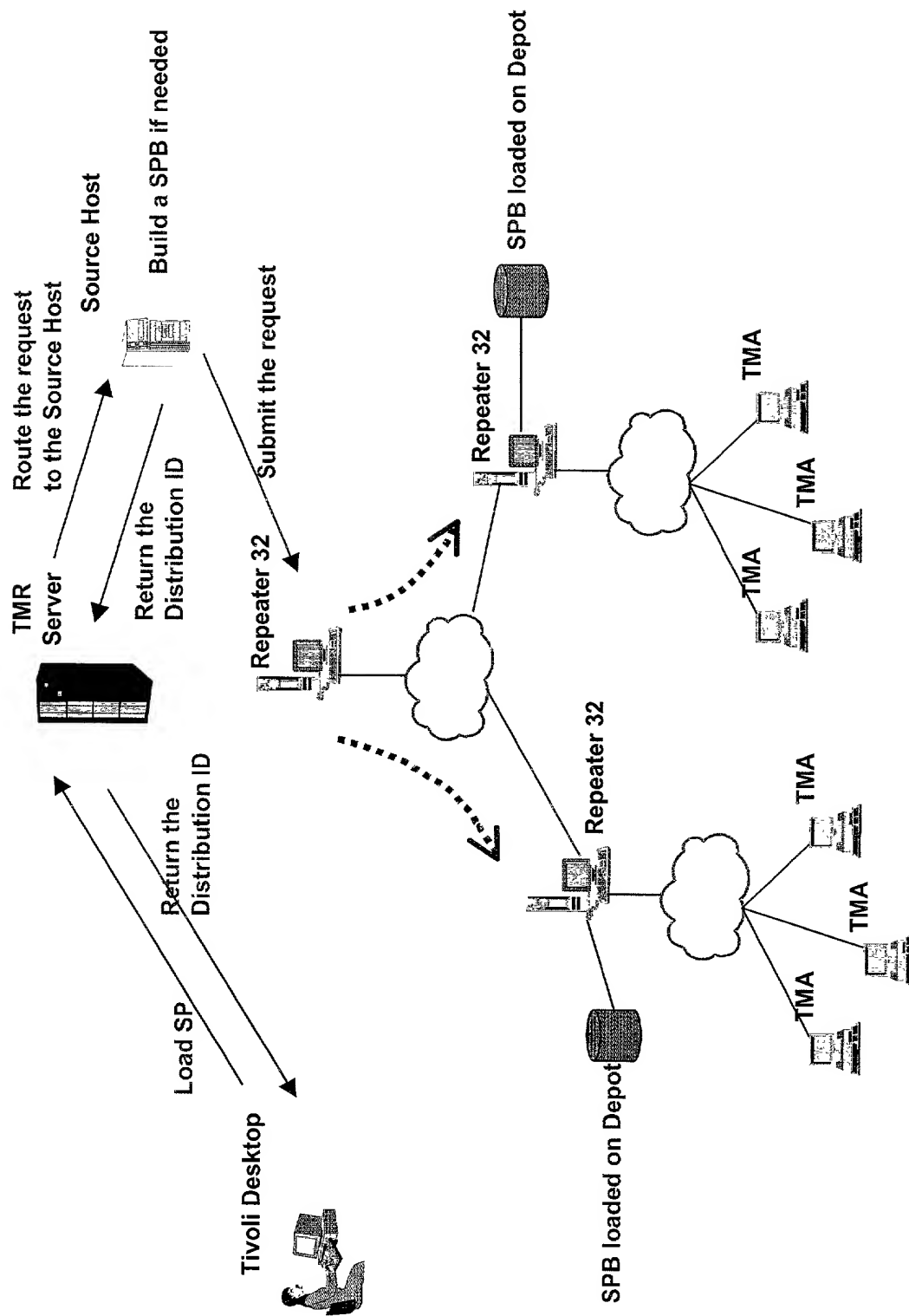|	
wimpspo	Associate a Software Package to a SPO
wconvspo	Convert the format of a SPO
wexpspo	Export a SPO into SPD file
wldsp	Loads SP on one or more Depots
wuldsp	UnLoad a SP from one or more Depots
winstsp	Install a SP
wremovsp	Remove a SP
wcommtsp	Commit a SP
wundosp	Undo a SP Install/Remove
waccptsp	Accept an undoable Install/Remove
wversp	Verify the state of a SP

Figure 16

Route the request
to the Source Host

Source Host

Submit the SD Operation

TMR
Server

Return the
Distribution ID

Repeater 32

Reports back to TMR Server

Method runs on TMAs

TMA

TMA

TMA

Submit a SD Operation

Repeater 32

Return the
Distribution ID

TMA

Tivoli Desktop

TMA

TMA

Method runs on TMAs

Figure 17

Build a SPB if needed

Source Host

Route the request
to the Source Host

SPB loaded on Depot

TMR
Server

Submit the request

Return the
Distribution ID

Repeater 32

Repeater 32

TMA

TMA

Return the
Distribution ID

TMA

Load SP

Repeater 32

TMA

Tivoli Desktop

TMA

TMA

SPB loaded on Depot

Figure 18

Figure 19

	Install	Remove	Accept	Undo	rollBack	Commit	Verify
transactional	X	X		X			
undoable	X	X					
undoable in transactional	X	X					
transactional & undoable	X	X					

Figure 19

Operation	State	Undo state	Reboot state	Flag
Install	Prepared	Prepared	ReBoot requested	Changing
Remove	Committed	Undoable	-	In Error
		Restored		-
		-		

IC--- means that an Install has been committed

ICU-- means that an Install has been committed and can be undone

IP-BC means that an Install has been prepared and it will be committed during the next reboot

RCU-- means that a Remove has been committed but it can be undone

IC--E means that an Install has been committed but the SP is in error, so the application may not work properly
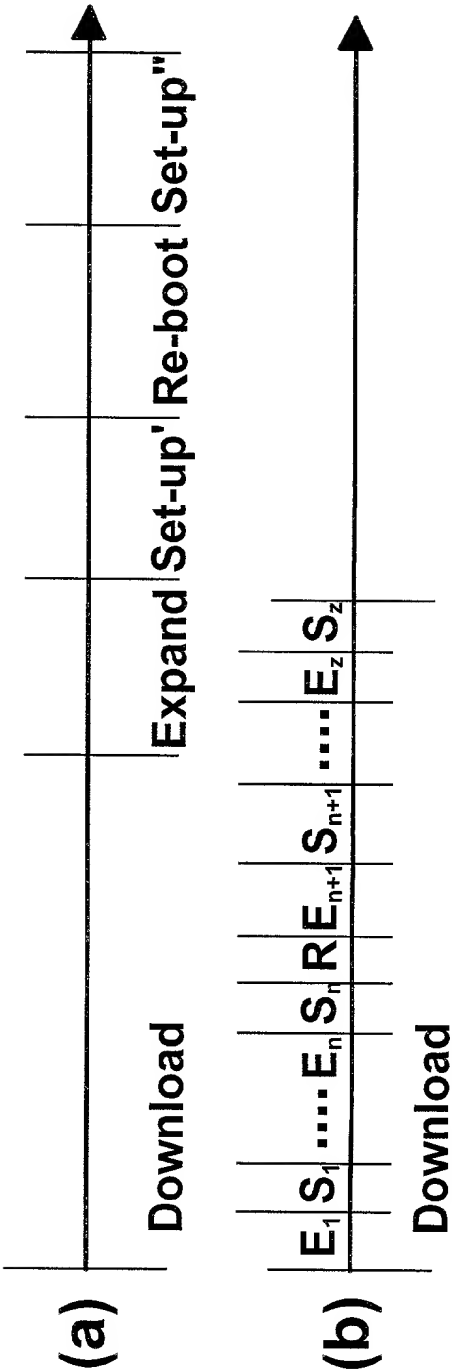
Figure 20

Set Checkpoints
periodically

TMA

Restart from last
Checkpoint

Repeater 32

Request data from last checkpoint

Receiving data...

TMA recalled

Figure 21

(a)

Download | Expand | Set-up | Re-boot | Set-up'' →

(b)

Download | E_1 | S_1 | .... | E_n | S_n | R | E_{n+1} | S_{n+1} | .... | E_z | S_z →

Figure 22

State after Remove

Remove

State before Install

State after Install

Figure 23

State after Undo

Undo

State before Install

State after Install
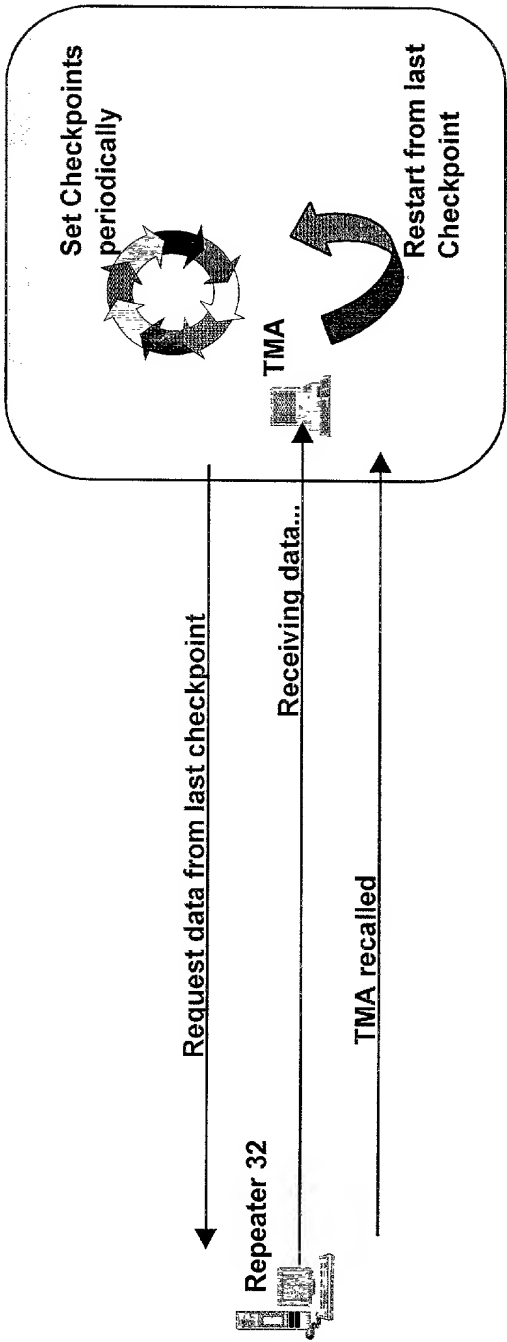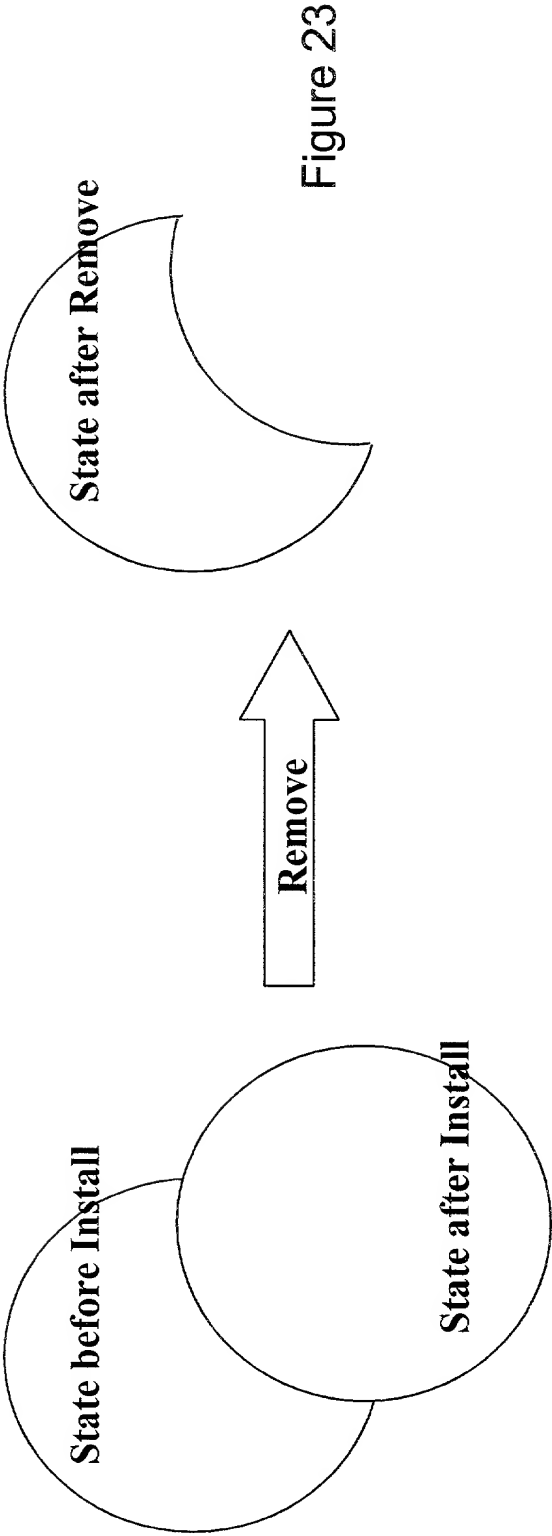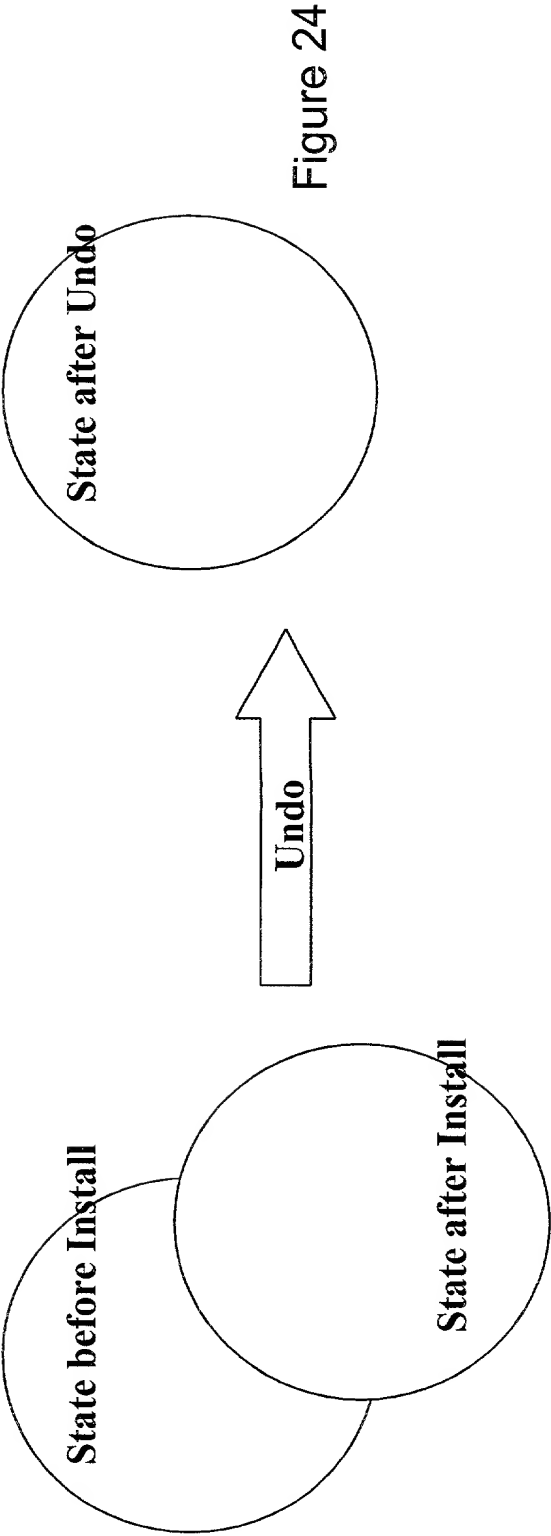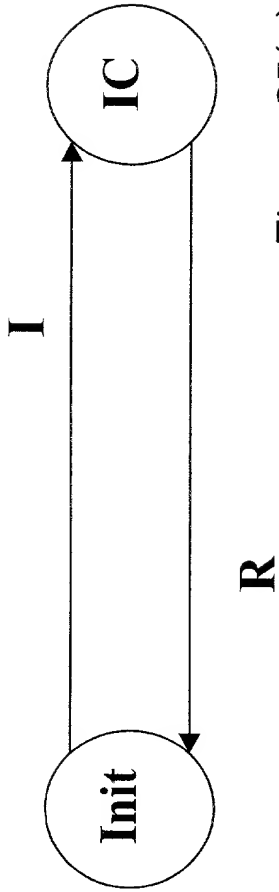
Figure 24
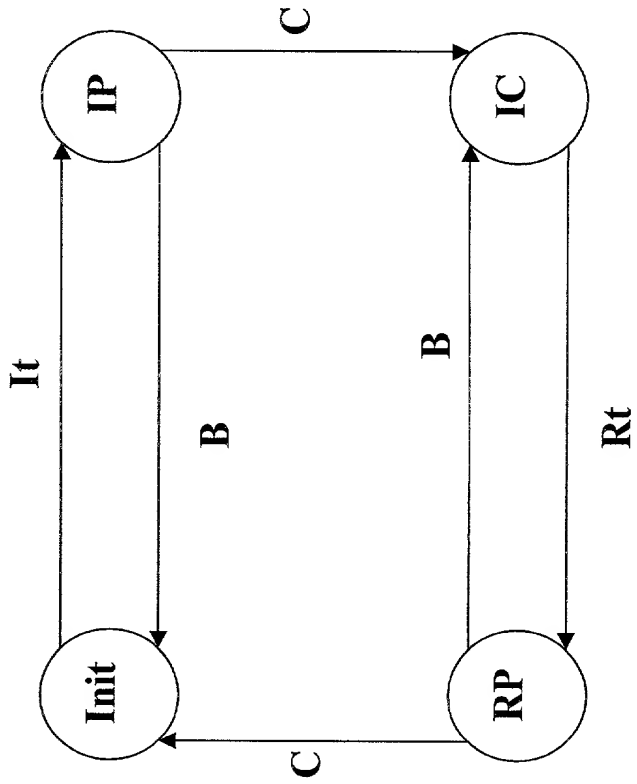
Figure 25(a)
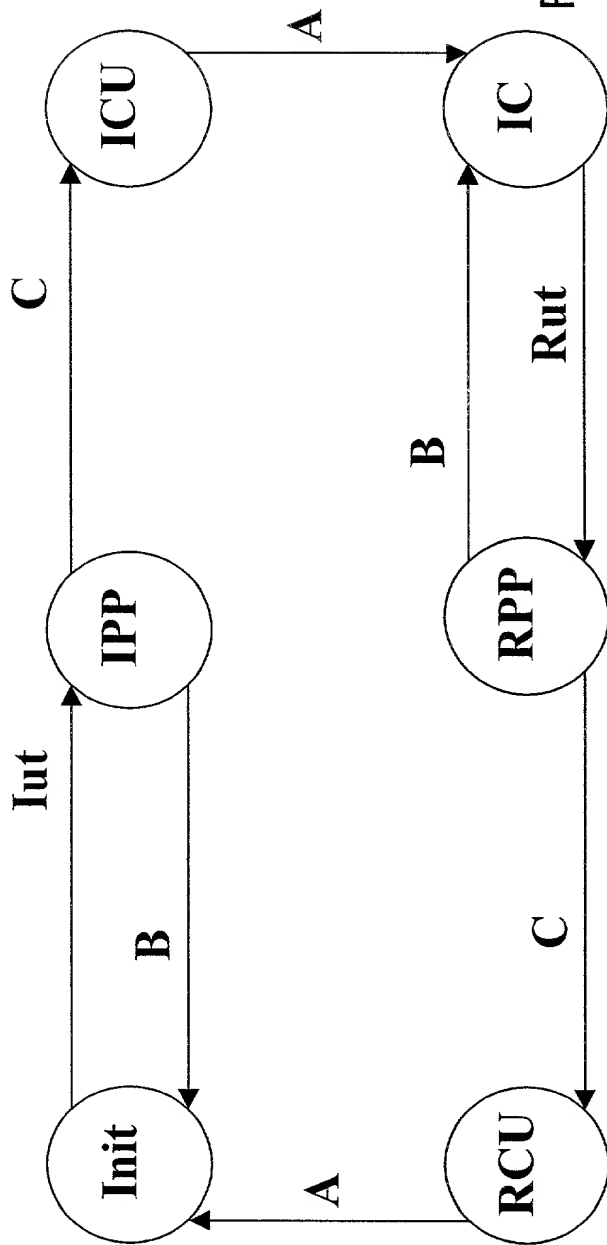

Figure 25(b)
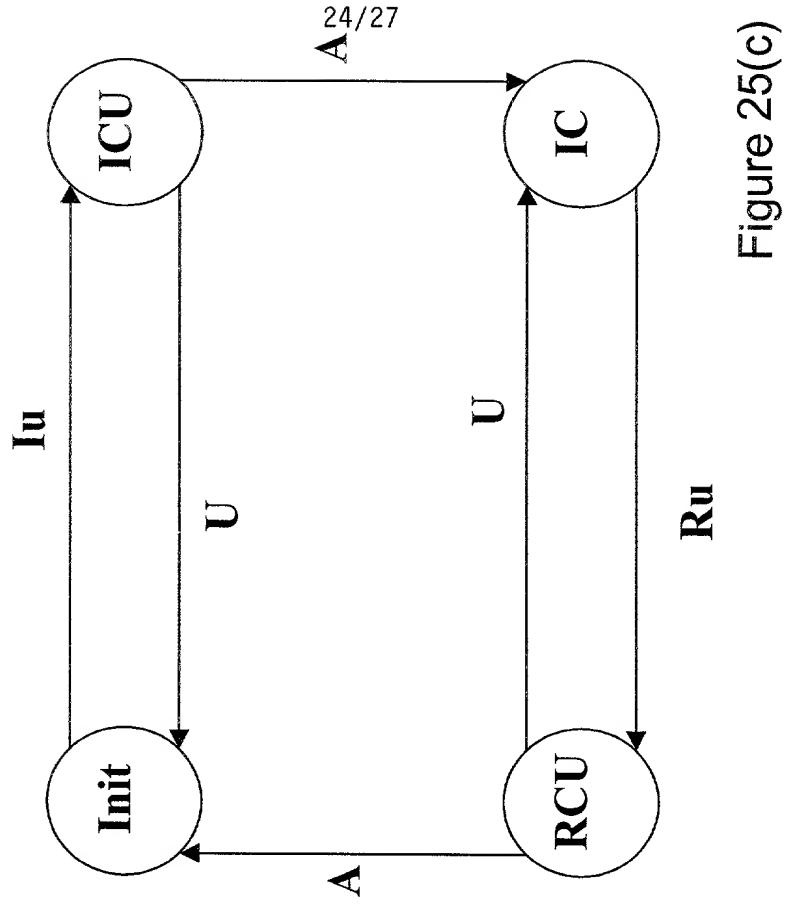
Figure 25(c)

Figure 25(d)
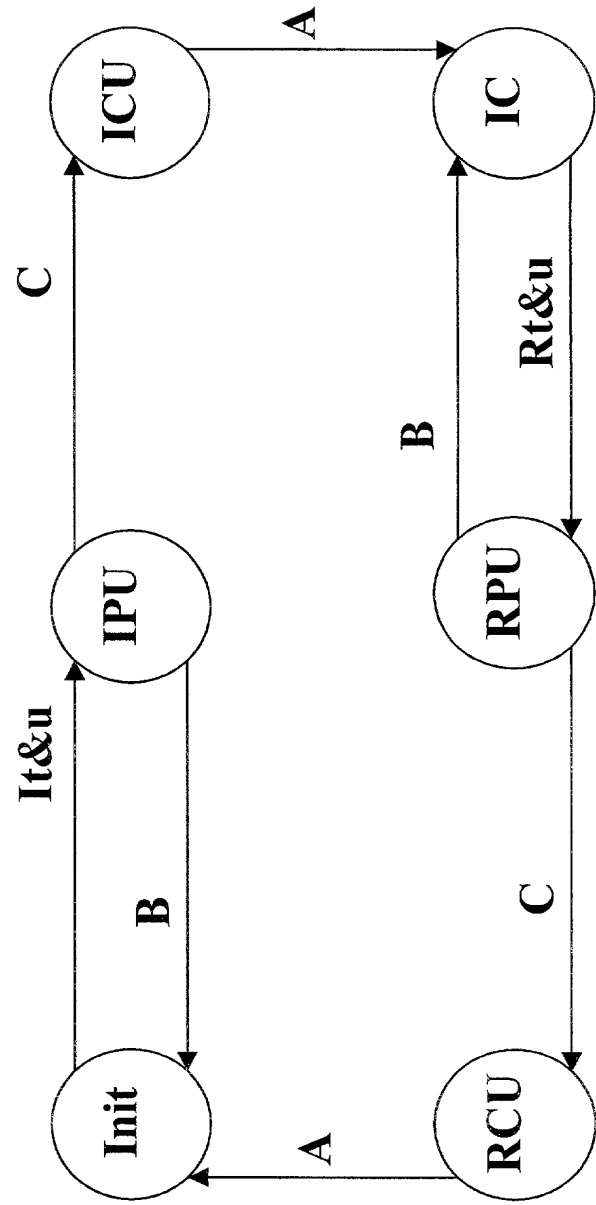

Figure 25(e)
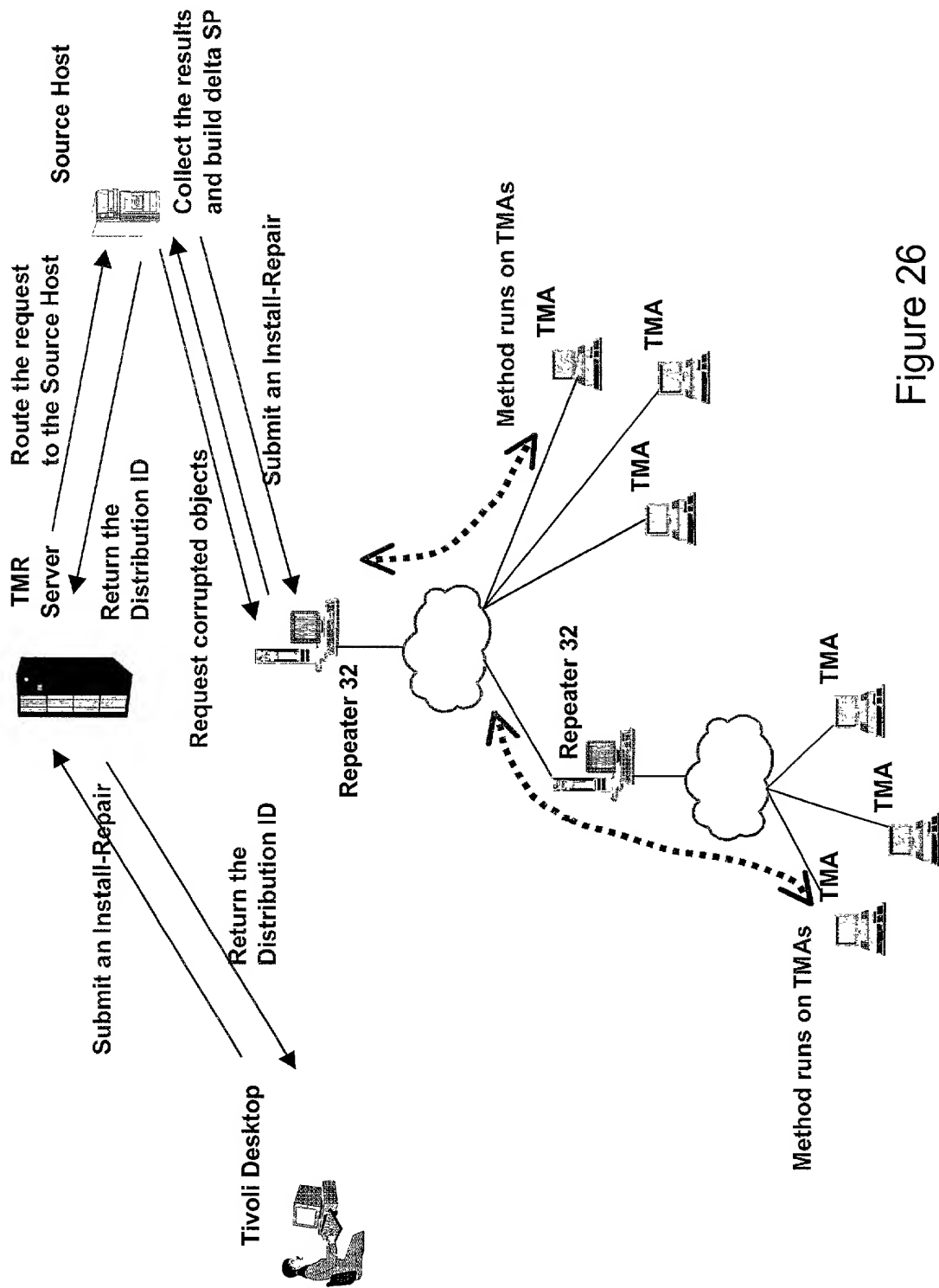
Figure 26

```
...
## Mount the image directory
execute_user_program
    during_install
        name      = $(system_dir)/NET.EXE
        arguments = "USE $(freedrive_01)  $(remote_name)"
    end
end
## Run the setup to perform the installation
execute_mssetup_program
    during_install
        name      = $(freedrive_01)/setup.exe
        arguments = "/Q1 /B1"
    end
end
## U-mount the image directory
execute_user_program
    during_install
        name      = $(system_dir)/NET.EXE
        arguments = "$(freedrive_01) /d"
    end
end
...
```

Figure 27